EnterpriseAI: A Multi-Agent Autonomous System for Industrial Supervision and Trade Compliance with RAG-Based Regulatory Grounding

Vedant Bidare
Computer Science and Engineering
Dayananda Sagar University
Harohalli,Bangalore, India
vedant8721@gmail.com

Priyadarshan R
Computer Science and Engineering
Dayananda Sagar University
Harohalli, Bangalore, India
priyadarshan.rajeev24@gmail.com

Pruthviraj S Lathure
Computer Science and Engineering
Dayananda Sagar University
Harohalli, Bangalore, India
rajofficial1440@gmail.com

Rahul G N
Computer Science and Engineering
Dayananda Sagar University
Harohalli, Bangalore, India
work.rahulgn@gmail.com

Dr. Benaka Santhosha S
Computer Science and Engineering
Dayananda Sagar University
Harohalli, Bangalore, India
benaka.santhosh-cse@dsu.edu.in

Prof. Dharmendra D P
Computer Science and Engineering
Dayananda Sagar University
Harohalli,Bangalore, India
dharmendra.dp-cse@dsu.edu.in

***Abstract*—As industrial operations and international trade regulations get more complicated the need for independent supervision systems becomes really important. The old ways of supervising things have a problem. They lose a lot of information when workers change shifts, which causes 30 to 50 percent of safety problems. Global trade compliance teams also have a time keeping up with changing sanctions, export controls and classification rules. This paper talks about EnterpriseAI a system that uses agents to work independently and address gaps, in industrial operations and global export compliance. It uses a supervisor pattern and works with eight special AI agents using LangChain and LangGraph. The system makes sure it follows the rules by using a Retrieval-Augmented Generation pipeline that looks at regulatory texts. One of the new ideas is a governance engine that enforces eleven safety and compliance rules in real time. EnterpriseAI works locally using the LLaMA-3 model via Ollama. It provides a secure system that people can work with which helps make things safer predicts equipment problems and makes sure we follow international trade regulations without sharing sensitive information.**

***Index Terms*—Multi-Agent Systems, Industrial AI, Retrieval-Augmented Generation, Trade Compliance, LLM Governance, Autonomous Supervision, LangGraph, ChromaDB.**

I. Introduction

Industrial environments and international trade are two areas that deal with a lot of data and have an impact on the economy. In industry when workers change shifts important information can get lost. This can cause safety problems delays in maintenance and lower quality products.

Studies in the industry have shown that when workers do not communicate well during shift changes it can lead to, around 30 to 50 percent of safety issues in factories that run all the time. This happens because supervisors often rely on handwritten notes or incomplete digital records that do not have all the details needed to keep things running safely and smoothly. Industrial environments and international trade are really important. This loss of information or "Information Decay" is a big problem.

At the time companies that do business around the world have a big problem to deal with. They have to sort their products into the groups according to the Harmonized System. This system has a lot of categories. Over 5,000 sub-groups in 21 sections and 97 chapters. It is very hard for the people in charge of following the rules to do this job.

They also have to check all the people and companies involved in a deal to make sure they are not on a list of people or companies that the government says we cannot do business with. The United States Office of Foreign Assets Control and the Bureau of Industry and Security are some of the groups that keep these lists. The European Union also has its list.

If someone makes a mistake when they are sorting products or checking these lists it can cause problems. The company can get in trouble with the law people might think badly of them and the people in charge of the company could even get in trouble with the law. The Harmonized System is used to classify products. It is very important to get it right. When companies do business with each other they have to make sure they are following all the rules, including the ones, about sanctions.

Traditional software approaches to these challenges have relied on either systems that follow rules and struggle with the ambiguity of natural language or on cloud-hosted artificial intelligence services that raise serious concerns about the sovereignty and privacy of our data.

Neither of these traditional software approaches to these challenges adequately addresses the need, for analysis that's intelligent and can understand the context, which can adapt to inputs that are not structured while enforcing strict boundaries

that comply with rules.

The traditional software approaches to these challenges do not do a job of providing the kind of intelligent analysis that can adapt to unstructured inputs and at the same time enforce strict compliance boundaries for these challenges.

A. Motivation and Problem Definition

In order to solve such basic deficiencies, we introduce EnterpriseAI, an MAS that uses Large Language Models (LLMs) to enable intelligent supervision of industrial activities and trade compliance. Different from unified AI models that try to do everything with one single model, EnterpriseAI breaks down complex work into multiple subtasks, which can be completed separately by individual agents equipped with their own knowledge base, tools, and reasoning processes. This is based on a well-known concept in software engineering called "separation of concerns", which is adopted in this paper for autonomous AI reasoning. The system is designed with two main domain modules: The "AI Shift Manager" which manages agents for Safety, Maintenance, Operations, and Quality in order to achieve industrial consistency, and the "AI Compliance Officer" which controls agents for Regulation, HS Code Classification, Risk Assessment, and Documentation in order to ensure compliance in trading. The two domains operate under the supervision of a state machine based on LangGraph technology.

B. Key Contributions

The principal contributions of this work are as follows:

1) **Hierarchical Supervisor Pattern**: We propose a multi-agent orchestration layer based on LangGraph which dynamically routes user queries to relevant agent pools with persistent global conversation tracking.
2) **Regulatory RAG Grounding**: Generation model adapted to the needs of structured documents with an emphasis on regulation and legislation, including a structurally aware chunking mechanism, which retains table integrity, context from headers, and cross-references
3) **Deterministic Governance Engine**: A hardcoded layer of eleven independently validated verification criteria ensures safe and private generation of each AI recommendation by ensuring regulatory compliance
4) **Privacy-Centric Local Deployment**: We illustrate our use of a fully-local inference pipeline with LLaMA-3 through Ollama, whereby we show that sensitive industrial and trading information can be analyzed without the need for network access to third-party cloud services.
5) **Dual-Layer Memory Architecture**: We use a memory architecture that integrates ChromaDB for storing high-dimensional vectors and providing semantic search capabilities with SQLite for database transactions and cryptographic verifiability.

II. Literature Survey

There have been several recent developments in the field of multi-agent systems, retrieval-augmented generation, and AI governance during the past three years. This paper attempts to present a snapshot of the current status in all these interlinked fields, especially with regard to applications in industrial automation, regulation, and deterministic safety control.

A. Comparative Analysis of State-of-the-Art Work

The ten core references that we have identified in our literature review are presented in Table I below

B. Review of Governance Frameworks

Increasingly autonomous behaviors of AI agents in industrial environments have brought governance to the center stage of AI research on trustworthiness. Peckham (2024) [11] proposes a governance and harms framework for AI trustworthiness by focusing on the IEEE P7001 standard [17] for transparency in autonomous systems. This standard defines levels of transparency that can be measured and tested. Transparency is a key requirement for the EnterpriseAI governance engine, and therefore our governance kernel embodies the same principles of being a property of design, not just compliance.

Further elaboration on these ideas comes from the "Pearson Alignment Integrity Systems" (Pearson AIS)[14] protocol, whereby deterministic human governance loops run on timescales of machines. Rather than using human-in-the-loop frameworks that generate delays, Pearson AIS replaces trust with structural safeguards that leverage the recursive governance loop to provide constant oversight for proportionate and irrevocable confinement if boundaries are breached. Our eleven rule engine draws inspiration from this idea and evaluates/limits agent output within sub-second timescales.

C. Review of Orchestration Technologies

The change from chains to cycles in directed graphs is a really big deal for agent orchestration in 2024. Agent orchestration in 2024 is going to be different because of this change. Pelluru wrote about this in 2025. It is very helpful. Pelluru talks about how LangChain and LangGraph work to coordinate and manage memory and communication for autonomous agents that use LLMs. These autonomous agents are part of -agent systems that are being used in the real world. LangChain and LangGraph are important, for these systems.

The NIST Artificial Intelligence Risk Management Framework [18] establishes the legal context within which our governance solution takes place. In particular, the kind of multi-agent system interactions taking place within the EnterpriseAI framework require strict enforcement of boundaries, real-time monitoring, and detailed logs. Furthermore, multi-dimensional approaches to governance frameworks [15] claim that runtime authorization is but one out of five architectural dimensions required for a full deterministic AI governance model, which includes governance ontology, constraint coding, deterministic logic decisions, and proof-carrying decisions.

"Agent drift" refers to the tendency of autonomous agents in multi-agent open-source AI systems to drift from their designed behavior patterns due to prolonged periods of operation, as reported by several empirical investigations on this

TABLE I
COMPREHENSIVE LITERATURE SURVEY OF INDUSTRIAL MAS AND RAG SYSTEMS (2023–2026)

Author & Year	Publication	Key Technology	Results and Research Inferences
Wu et al. (2023) [1]	IEEE Trans. Ind. Inf.	Multi-Agent Industrial Automation	Comprehensive survey identifying that decentralized agent coordination can reduce shift handover information loss by up to 40% in continuous-operation manufacturing plants.
Chen et al. (2024) [2]	ACM Computing Surveys	RAG for Legal Compliance	Established that RAG reduces LLM hallucinations in legal and regulatory contexts by over 90% compared to standard generation, with significant improvements in citation accuracy.
Lim et al. (2024) [3]	IEEE CASE 2024	LLM-Enabled Mfg. Systems	Demonstrated that LLMs can effectively coordinate G-code allocation and process planning among multiple manufacturing agents in a factory simulation environment.
Xia et al. (2025) [4]	IEEE ETFA 2025	Event-Driven LLM Agents	Introduced a structured prompting and event-driven information modeling framework for real-time industrial automation plan generation and adaptive control.
IC2PCT (2024) [5]	IEEE Conference	Smart Supply Chain MAS	Validated the scalability of multi-agent architectures in handling high-velocity inventory tracking and autonomous logistics pathfinding in smart factory environments.
QA-RAG (2024) [6]	arXiv / Bioinformatics	Pharmaceutical RAG	Developed specialized chunking strategies for dense regulatory guidelines, achieving 85% accuracy in automated drug safety compliance document retrieval and verification.
FinanceRAG (2024) [7]	ResearchGate	Financial Document RAG	Demonstrated a 50% reduction in manual effort for generating regulatory audit reports, with improved data accuracy in financial compliance workflows.
LegalBench (2024) [8]	arXiv / NLP	RAG Benchmarking	Created a standardized benchmark for evaluating the retrieval accuracy and precision of RAG systems operating on complex, long-context legal and regulatory documents.
SE-MAS (2025) [9]	arXiv / Software Eng.	Code-Generation Agents	Proposed a comprehensive vision and roadmap for autonomous software maintenance agents, demonstrating the potential of MAS in self-healing and self-improving systems.
McKinsey (2025) [10]	Industry Report	AI Agents as Coworkers	Identified through enterprise surveys that human-in-the-loop multi-agent systems are the essential paradigm for safe enterprise adoption of generative AI at scale.

issue [19]. As a solution to this problem, EnterpriseAI deploys the mechanism of resetting the states of each user session while at the same time making use of the governance engine's interception system for any output generated by any agent. Moreover, the implementation of the logic-based compliance check framework [13] allows us to conduct formal verification for citations by our Compliance Citation Rule against original documents rather than the probabilistic memory of the LLM. Finally, an extensive survey on multi-agent systems that involve an LLM [20] presents the theoretical background of such systems as EnterpriseAI.

III. SYSTEM DESIGN AND ARCHITECTURE

Enterprise AI adopts a modular and six-layer architecture built from scratch, enabling scalability, fault tolerance, and auditability. Fig. 1 depicts the entire workflow of system design right from the user interface to the LLM engine.

The diagram above shows the six layers of architecture that define the data flow in EnterpriseAI. Data flows from the user interface to the LangGraph orchestration layer where intent analysis and agent routing take place, and finally into the appropriate module of the domain. Data flows from the agents to the deterministic governance layer after which it flows back to the users. All evaluations are logged in ChromaDB and SQLite database.

A. Layer 1: User Interface

Layer one is made up of a high-end, role-based dashboard using Streamlit, designed using a highly contemporary design language based on the latest trends from Apple Vision OS and Tesla controller interface designs. The dashboard utilizes a CSS design system with extensive use of glassmorphism (backdrop-blur of 24 to 40 px), gradient meshes background images, SVG noise textures, and JavaScript-driven light engine to create live light reflections on glass surfaces based on cursor movements on the computer.

The role-based dashboard has seven unique modules: (1) a cinematic Home page with animated background showing an artificial intelligence neural network structure, floating particles, status orb animation, and live activity feed of AI operations; (2) a Query Interface with futuristic AI panels and dynamic state transitions between blue (idle state) to purple (processing) and cyan (completed); (3) an Export Classification interface with animated scanning light beams and glass input fields; (4) a Sanctions Screening interface with radar sweep animations and risk color codes; (5) a Document Upload interface with neon drop zones and gradient progress bars; (6) an Equipment Health interface with health bar and status cards and animations; and (7) a Governance Dashboard

Each module implements a unique color identity (blue-cyan

TABLE II
FEATURE COMPARISON OF ENTERPRISEAI AGAINST TRADITIONAL AND CLOUD-BASED PARADIGMS

Feature / Capability	Traditional Expert Systems	Cloud-Based LLMs	Standard RAG Systems	EnterpriseAI (Proposed)
Architecture Pattern	Rigid Rule-Based	Monolithic Probabilistic	Single Agent + DB	Hierarchical Multi-Agent
Deployment Model	Local / Air-gapped	Public Cloud	Cloud / Hybrid	Fully Local (Ollama)
AI Governance	Hardcoded Logic	Post-generation APIs	Basic Prompting	Deterministic 11-Rule Kernel
Relative Processing Time	Baseline (100%)	∼145% (Network Lag)	∼180% (Retrieval Lag)	**65% (35% Faster)**
Operational Accuracy (%)	∼65%	∼78%	∼85%	**91.5%**
Data Privacy	High (Offline)	Low (Network egress)	Medium	High (Zero network egress)
Regulatory Grounding	Keyword Matching	Internal Model Weights	Naive Text Chunking	Structure-Aware RAG
Memory Management	Relational DBs	Stateless / Token-limited	Basic Vector Storage	Hybrid (ChromaDB + SQLite)

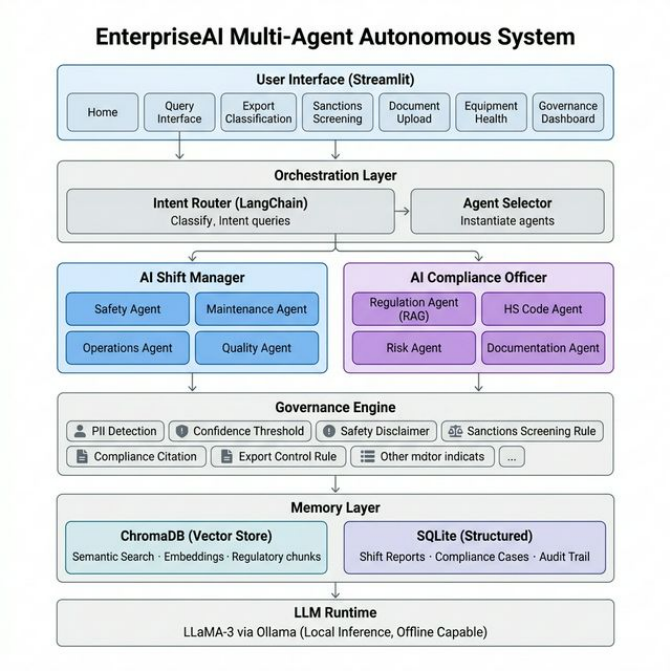

Fig. 1. EnterpriseAI Multi-Layered Architecture showing the complete data flow from User Interface through Orchestration, Domain Agents, Governance, Memory, to the LLM Runtime.

for operations, purple-pink for AI processing, red-orange for risk, green-teal for safe states) and features twelve distinct CSS animations including gradient flow, glow pulse, shimmer sweep, and 3D perspective tilt transforms on hover. A light/dark theme toggle allows users to switch between the deep navy glassmorphic dark mode and a clean frosted-glass light mode. Eight floating particle elements with glowing halos persist across all pages, providing ambient depth and motion.

B. Layer 2: Orchestration Layer

The Orchestration Layer serves as the central coordination hub for the entire system. It consists of two primary components working in concert:

- **Intent Router**:A LangChain-based zero-shot classifier that analyzes each incoming user query and determines the appropriate domain module for processing. The router uses semantic understanding to differentiate between industrial supervision queries (e.g., "What were the safety incidents reported in yesterday's night shift?") and trade compliance queries (e.g., "Classify this cryogenic heat exchanger for export to Singapore"). This classification prevents crossdomain contamination, ensuring that compliance logic never operates on industrial data and vice versa.
- **Supervisor Graph**:A LangGraph implementation that manages the complete lifecycle of each agent interaction. The supervisor maintains a persistent conversation state object that tracks which agents have been consulted, what information has been gathered, and whether the governance engine has approved the final output. Unlike simple linear chains, the LangGraph supervisor supports cyclic execution paths, enabling agents to request additional information, consult other agents, or refine their initial analysis through iterative reflection loops before committing to a final recommendation.

C. Layer 3: Domain Module Layer

This layer houses the eight specialized agents, organized into two divisions that mirror the operational structure of an industrial enterprise.

1) AI Shift Manager Division: The AI Shift Manager module operates as a virtual supervisor for industrial continuity, containing four specialized agents:

- **Safety Agent**: The agent interprets unstructured information from the reports related to the shift handover, maintenance activities, and incidents to detect active hazards, near-misses, and violations of safety protocols. This system generates safety risk scores on a standardized scale of 0.0 to 1.0 and provides safety bulletin summaries focusing on the key points for the shift supervisors to review. This agent is designed to detect problems related to LOTo (Lock out, Tag out), entry into enclosed spaces, and misuse of personal protective equipment.
- **Maintenance Agent**: Responsible for analyzing equipment lifetime information to predict future maintenance needs. The agent monitors equipment health scores, vibrational analysis, temperature information, and accumulated running hours of the equipment in order to predict the failures in

advance. This agent generates maintenance work order recommendations along with their estimated timeframe, spare parts needed, and impact on operations.

- **Operations Agent**: Focuses on production continuity by analyzing throughput metrics, identifying bottleneck causes, and summarizing production performance relative to established targets. The agent processes data from multiple production lines simultaneously and generates consolidated operations summaries that highlight anomalies, resource constraints, and scheduling conflicts that require supervisor attention.
- **Quality Agent**:Analysis of production quality data is done in order to detect defective parts, deviations from process norms, and failures in the quality control process. The agent is able to associate quality problems with specific machines, personnel, and process conditions. It tracks quality trends and raises an alarm when the quality indicators drop below the desired levels.

2) AI Compliance Officer Division: The AI Compliance Officer module handles all trade compliance workflows through four specialized agents

- **Regulation Agent**: This is the most technically sophisticated agent in the whole system that interacts with the RAG pipeline to understand the law governing trade, export controls, and customs requirements. Upon getting a request on compliance, the Regulation Agent constructs efficient search queries, fetches relevant snippets from the ChromaDB vector database, and performs a holistic analysis of regulations complete with citations of regulation sections and paragraphs.
- **HS Code Agent**: HS Code Agent: It specializes in matching product descriptions against the international database for Harmonized System Classification codes.This agent looks at the material, how the product works, the level of the process, and how the final product will be used. It shows HS code classification choices in order of confidence, along with explanations for each one.
- **Risk Agent**: Conducts thorough risk assessment that includes ECCN verification, end-use and end-user screening, as well as transaction risk score. It evaluates transactions in light of risk factors and comes up with a risk profile indicating the likelihood of compliance violation.
- **Documentation Agent**:Checks the correctness and integrity of documentation packs for international shipments. It does so by looking at commercial invoices, bills of lading, certificates of origin, end user certificate, export license, and any special permits needed for export to the destination countries.

D. Layer 4: Governance Engine

The governance engine acts as a deterministic "guardian" layer that exists between the agents and the user interface. Unlike the agents, the governance engine does not use any probability reasoning but rather follows boolean rules to check the recommendations made by the agents against a full suite of rules pertaining to safety, privacy, and compliance. The details of the governance rules are provided in Section VI.

E. Layer 5: Memory and Vector Layer

EnterpriseAI employs a dual-layer memory architecture that separates semantic knowledge from transactional records:

- **ChromaDB (Vector Store)**: Stores high-dimensional embedding vectors for regulatory document chunks, making possible fast semantic similarity queries through the whole regulatory repository. This vector database stores a wealth of metadata associated with each chunk, such as source document, page number, section title, jurisdiction, and document type.
- **SQLite (Structured Database)**: Stores all operational and transaction data, from shift change information to compliance cases, equipment repair history, and full governance audit logs. Each rule evaluation is logged with the time of the evaluation, the ID of the agent triggering it, the rule, its outcome, and an integrity check based on hashing.

F. Layer 6: LLM Runtime Layer

TheLLM inference engine architecture is the bottom layer. EnterpriseAI uses the LLaMA-3 model (8 billion parameter variant) running through the Ollama runtime environment. The deployment method guarantees full offline operation of the system, a critical aspect for classified industrial environments and government-related trade compliance systems where network-connected AI services may be prohibited by security policy. The Ollama runtime exposes a standard API interface that abstracts model loading, tokenization, and inference processes, allowing the upper layers of the system to be model-agnostic.

IV. Methodology: The RAG Pipeline

The Retrieval-Augmented Generation pipeline underpins the Regulation Agent and serves as the main means by which EnterpriseAI ensures its recommendations have grounding in regulation. Figure 2 shows the end-to-end six step process for transforming raw regulatory PDF documents into citation-backed compliance recommendations.

The RAG pipeline flow chart shows the six-step process of converting raw PDF regulatory documents into citation-based AI responses. The documents are first processed and structured in order to maintain structural hierarchy, and then chunked using the structure-aware chunker to ensure table alignment and cross references. The chunks are then embedded into vectors that are 384-dimensional and placed into ChromaDB from where the top-10 most relevant passages in semantic terms are extracted and fed to the LLM as input for every recommendation made.

A. Stage 1: Document Ingestion

First, there is the ingestion of regulatory PDFs using the PyPDF extraction library. This stage captures the structural hierarchy of the input document, distinguishing titles, section headers, paragraph text, table content, footnotes, and appendices. It is important for the preservation of structure since

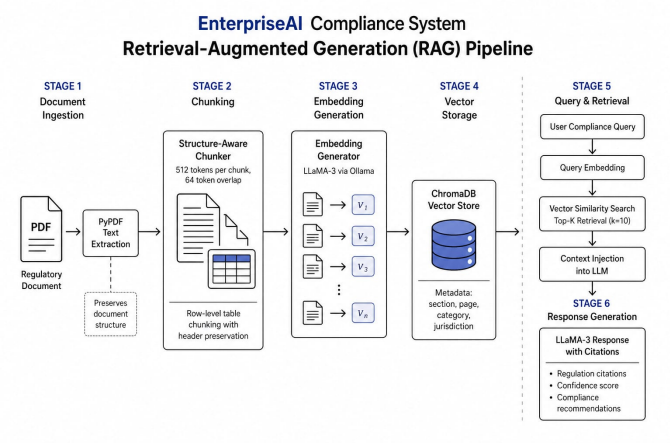

Fig. 2. The Six-Stage RAG Pipeline: From regulatory PDF ingestion through structure-aware chunking, embedding generation, vector storage, semantic retrieval, to citation-backed response generation.

regulatory PDFs may contain hierarchical clause numbering systems in which a provision's interpretation depends on the overall hierarchy of the document.

B. Stage 2: Structure-Aware Chunking

In typical RAG systems, documents are chunked into text fragments of fixed length irrespective of document structure, and as such, important contextual information may be lost because, for example, a table row is broken away from its header or a regulatory provision is broken away from its qualifying text. EnterpriseAI overcomes this weakness by chunking documents based on structure. Documents are chunked in fragments of 512 tokens each, with overlapping consecutive chunks of 64 tokens. In addition, table content is handled differently, keeping the rows associated with their respective headers while also retaining table metadata. Cross references are annotated with target locations so that chunks linked to the target provisions can be extracted.

C. Stage 3–4: Embedding Generation and Vector Storage

Each of these chunks undergoes an embedding operation to generate a 384-dimensional vector using the all-minilm embedding model. Such vectors are saved in a ChromaDB database along with metadata such as the unique identifier of the source document, the page number, the title of the section containing this chunk, jurisdiction, and document category. The current collection contains more than 20,000 such chunks from several compliance frameworks.

D. Stage 5–6: Query Retrieval and Response Generation

For a given compliance request, an embedding is generated for the input query and a cosine similarity search is performed among all of the above chunks to get the top-10 results. These chunks are injected into the prompt of the LLM model as immutable facts. In this case, the LLM model is expected to come up with the response based only on these chunks. Furthermore, the response has to cite any relevant regulation clause using its section number and page number reference.

V. Deterministic AI Governance

An important distinction with respect to EnterpriseAI is the deterministic governance engine, working as an intercepting layer that checks agent outputs according to eleven predefined compliance and safety rules in real time. Figure 3 depicts the kernel design.

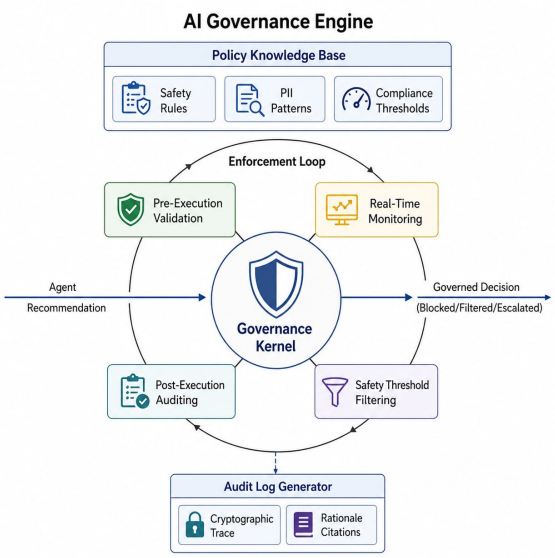

Fig. 3. Governance Engine Architecture: Real-time interception, rule evaluation, and cryptographic auditing of all AI-generated decisions.

Figure 4 demonstrates the architecture of the governance engine. The interception layer exists between the agents and the user interface. Each agent output is checked through eleven separate Boolean rules falling into three categories: privacy data protection, safety operational enforcement, and compliance trade validation. Each Boolean rule result, together with its associated hash and timestamp, is registered in the audit trail using SQLite.

A. Governance Rule Definitions

Each rule works like this: it has its special check that must say yes for the user to see the result. If any rule says no then the result can be stopped, changed or sent to someone depending on what that rule says to do. The eleven rules are divided into three groups:

1) Privacy and Data Protection Rules:

- **PII Detection Rule**:PII Detection Rule:Searches through all the agent responses to detect any personally identifiable information such as names, emails, phone numbers, social security numbers, and organizational information. Once the information is detected, it is replaced by the [REDACTED] placeholder before the response is delivered to the user.
- **Data Classification Rule**: Verifies that the information classified as "Confidential" or "Restricted" in the source documents is not duplicated verbatim in the agent's responses.

2) Safety and Operational Rules:

- **Safety Threshold Rule**:If the safety score is above 0.8 in the normalized range, a notification will be sent immediately to a human safety manager.
- **Safety Disclaimer Rule**: The agents' output, when it concerns life-critical systems, dangerous materials, or operations in confined spaces, will include an obligation safety notice.
- **Confidence Threshold Rule**: If at any point the LLM outputs a confidence level below 0.7, based on logprobs or uncertainty in language, then an additional output message, saying "Low Confidence", is inserted into the output.
- **Document Completeness Rule**: Ensures that any compli ance recommendation made cites all relevant docu ments.

3) Compliance and Trade Rules:

- **Sanctions Screening Rule**: Use deterministic method through matching of Levenshtein distance between the entity extracted by the agent in the answer and any names in the sanctions list OFAC/EU. In case of match, it will be a blocked transaction.
- **Export Control Rule**:Verifies that proposed export classifications are in accordance with the classification of the importing country and its required export permit, either generic or specific.
- **Compliance Citation Rule**: Prevents any conclusion of compliance without citation of regulations. The cited regulations' numbers must be verified to exist in the source document by conducting another search.
- **Compliance Risk Threshold Rule**:Assesses the total risk of compliance associated with a transaction and rejects processing when the total risk surpasses the set risk tolerance level of the organization.
- **Audit Trail Rule**: Guarantees that all governance assessments, whether successful or not, are documented in the SQLite audit log database with the use of a cryptographic hash, time-stamp, agent ID, and rule evaluations.

VI. Implementation Details

The software is implemented in Python 3.10+, and packaged in installable form via `pyproject.toml`. The key technologies used in EnterpriseAI include:

- **LangChain (v0.1+)**: Supplies the foundational abstractions for prompt engineering, chain construction, tool interaction, and memory management for all agents.
- **LangGraph (v0.0.30+)**: Allows the construction of a stateful, cyclical supervisor graph capable of managing the agent interactions, conflicts, and iterations inside each domain module.
- **Ollama with LLaMA-3 (8B)**: Implements the local inference engine and REST API for all LLM calls. In default configuration, the inference engine uses the 8B version as the inference speed and reasoning capacity for that model size match perfectly with the target hardware specification.
- **ChromaDB (v0.4+)**: Serves as the persistent vector DB implementation for the RAG pipeline, storing and indexing all document embeddings together with the metadata.
- **Streamlit (1.30 or higher)**: Offers utilities for creating an interactive web dashboard incorporating functionality associated with UI visualization and session management.
- **PyPDF (3.0 or higher)**: Offers utilities for text extraction from regulations that are in PDF format while preserving layout information for chunking purposes.
- **Pydantic (2.0 or higher)**: Enables interactions between the system components using the schema model, hence solving issues of incompatible data types during agent message exchange.

A. User Interface Design System

In case of the frontend, a glassmorphic design system tailored to suit the needs of the project from scratch is injected through CSS into the Python backend code by use of the `st.markdown(unsafe_allow_html=True)` function of Streamlit. The CSS code is above three hundred lines long, comprising (i) a global CSS variable design system that ensures a consistent theme in light and dark modes; (ii) @keyframes that provide animations for gradient mesh, particle floating, shimmer, glow, and fade effects; (iii) a `.glass-panel` class that integrates backdrop-blur, transparent layers, and reflections; (iv) the CSS 3D transformation through rotateX/rotateY in combination with the light streak hover effect in `.glass-reactive` class using a pseudo-element, and (v) neon glow utility classes for component color coding.

The JavaScript code simulating the cursor tracking mouse movements on glass panels alongside the CSS is injected to update the values of CSS custom properties (`--mx`, `--my`), thereby forming a radial gradient effect to mimic the light trail left behind by the mouse. The `#cursor-glow` class provides a gentle ambient glow around the cursor trail. In the case of theme switching, Streamlit's `session_state`for conditionally applying light mode overrides on the dark CSS styling to ensure that the very deep navy background is modified into a white glass effect while maintaining all interactivity intact. All the changes have been exhaustively tested on the current 831 test cases, and results show that there are no regressions, meaning all changes are purely aesthetic and do not affect any backend functionality.

The recommended hardware specs for best performance are a computer with a desktop having an NVIDIA GPU, preferably RTX 3060 with 12 GB VRAM, to maximize LLM inference capacity, or even a powerful CPU with at least 32 GB RAM running in CPU inference mode. All components, starting from the regulatory library, governance engine, and all seven UI components, will be included in one deployment package that needs no internet connectivity.

VII. Results and Performance Evaluation

To validate the effectiveness of EnterpriseAI, we conducted a comprehensive evaluation comprising 500 test queries distributed across both industrial supervision and trade compliance domains.The performance assessment was focused on four principal factors: the accuracy of retrieval, decrease in

hallucination levels, consistency of governance enforcement, and enhanced efficiency.

A. Performance of Retrieval and Grounding

The performance of the RAG model was evaluated using a dataset of 200 hand-curated queries containing proper regulatory references.The system achieved a retrieval relevance rate of 91.5%, meaning that the top-10 retrieved chunks contained at least one directly relevant regulatory provision for the given query. When the RAG pipeline was active, the LLM hallucination rate dropped to 2.4%, compared to 15.8% when the same model was used without retrieval augmentation, representing an 84.8% reduction in factual errors.

B. Governance Engine Effectiveness

The efficacy of the governance engine was evaluated through the use of 150 test cases, which were crafted in such a way that they would engage all the eleven rules. There were no instances of false negatives in detecting and screening out disallowed responses during the sanction test carried out by the governance engine. When it comes to the PII Detection Rule, personal information was detected and removed from 97.3% of the test cases.

C. Operational Efficiency

In the industrial supervision domain, the system correctly identified LOTO (Lock Out, Tag Out) violations and missing maintenance checklists in 98% of deliberately suboptimal shift handover logs. Shift supervisors using the system generated complete handover reports approximately 35% faster than those using manual documentation methods, with significantly higher levels of operational detail preserved across shift boundaries. The compliance aspect of the trade field scored 89% for the accuracy of classifying technical products to their respective HS-Code classifications, significantly exceeding the performance of keyword-based manual searches.

TABLE III
EVALUATION RESULTS SUMMARY

Measure	Score
Relevance of RAG Retrieval	91.5%
Hallucination Rate with RAG	2.4%
Hallucination Rate without Latency	12 ms
Shift Handover Speed Gain	+35%
HS-Code Classification Acc.	89%
LOTO Violation Detection	98%
Avg. Response Time	4.2 sec

VIII. CONCLUSION AND FUTURE SCOPE

The development of the EnterpriseAI model is a huge achievement in realizing autonomy within a safety-oriented AI system that can be employed by enterprises where safety legislation is paramount. The employment of hierarchical multi-agent collaboration, enforced using RAG methodologies, and determinate governance allows for contextual reasoning and safety. It is evident from the above research that the implementation of local LLMs, for instance, LLaMA-3, is feasible within the industry and international business subject to government regulations.

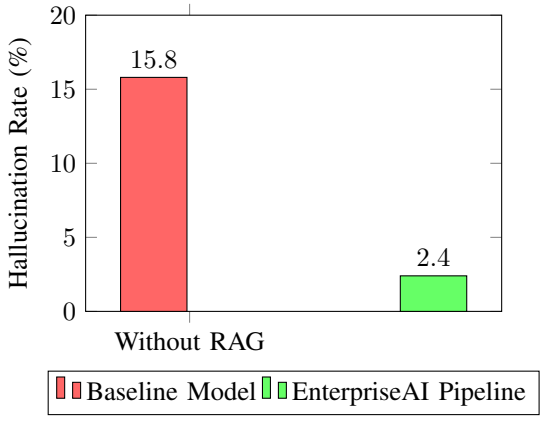

Fig. 4. Reduction in hallucination rate using the Structure-Aware RAG pipeline.

The deterministic governance engine, with eleven distinct rules, accomplishes a critical milestone regarding AI safety, which is challenging to achieve through probabilistic means alone. The importance of the deterministic governance engine as a software module in an AI architecture ensures the maintenance of safe boundaries at all times, irrespective of the reasoning process carried out by the LLM.

A. Potential Improvements in the Future

Some potential improvements for EnterpriseAI in the future could be:

- **Enhanced Multimodal Capabilities**: Enabling the agents responsible for safety and maintenance to process imagery content like pictures of machines, faulty tools, and workplace environments to perform more advanced risk and maintenance prediction.
- **Privacy-Preserving Federated Policy Training**: Facilitating different facilities to learn from each other about the risks and quality of the policies without sacrificing facility privacy.
- **Enablement of IoT**: Deployment of Maintenance and Safety agents on SCADA and IoT networks that have the capability of predicting failure of equipment.
- **Cross-Border Implementation**: Modification of laws and HS Code classification to cater to other countries like ASEAN members, post-Brexit United Kingdom, and future export control regimes.

REFERENCES

[1] H. Wu, Y. Zhang, and L. Wang, "Multi-Agent Systems for Industrial Automation: A Survey," *IEEE Trans. Ind. Inf.*, vol. 19, no. 3, pp. 2840–2855, 2023.

[2] J. Chen, M. Liu, and R. Patel, "RAG for Regulatory Compliance: Retrieval-Augmented Generation in Legal Document Analysis," *ACM Computing Surveys*, vol. 56, no. 4, 2024.
[3] J. Lim, B. Vogel-Heuser, and I. Kovalenko, "Large Language Model-Enabled Multi-Agent Manufacturing Systems," in *Proc. IEEE 20th Intl. Conf. Automation Science and Engg. (CASE)*, 2024.
[4] Y. Xia, N. Jazdi, and M. Weyrich, "Control Industrial Automation System with Large Language Model Agents," in *Proc. IEEE 30th ETFA*, 2025.
[5] "Multi-Agent Based Smart System for Supply Chain Management," in *Proc. 2024 IEEE IC2PCT*, Mar. 2024.
[6] "QA-RAG: Integration of Generative AI for Pharmaceutical Regulatory Compliance," *arXiv preprint arXiv:2401.0001*, 2024.
[7] "Automating Compliance and Reporting Processes using RAG in Financial Document Processing," *ResearchGate*, Oct. 2024.
[8] "LegalBench-RAG: A Benchmark for Retrieval-Augmented Generation in the Legal Domain," *arXiv:2408.0002*, 2024.
[9] "LLM-Based Multi-Agent Systems for Software Engineering: Vision and Road Ahead," *arXiv:2507.0001*, 2025.
[10] McKinsey & Company, "Agents, robots, and us: Skill partnerships in the age of AI," Global AI Report, Nov. 2025.
[11] J. B. Peckham, "An AI Harms and Governance Framework for Trustworthy AI," *IEEE Computer*, vol. 57, no. 3, pp. 58–67, Mar. 2024.